Employee Attrition Prediction using Machine Learning Algorithms

Abstract

In any corporation, if a significant number of employees leave their job with a short notice period, it may lead to a reduction in overall throughput which in turn will certainly have an impact on the turnover. Companies need to spend additional efforts in terms of time and cost to fill up the vacant position without any substantial loss to the ongoing business. To avoid these situations, we can use machine learning techniques to predict employees who are planning to leave the company with the help of some related data. One more way is to identify the features which inspire employees to leave their job. Refining such features in the company also will result in reducing the employee attrition rate of the company. In this paper, we attempted to predict employee attrition rate using the classification algorithms, Decision tree, Random Forest, K Nearest Neighbourhood, Neural Networks, eXtreme Gradient boosting and Ada-Boosting. We also have applied regularization for every algorithm to find the precise parameters to predict the employee's attrition rate considering the HR data set from Kaggle website which consists of 35 features with 34 independent and one dependent feature which is our attrition feature with Yes/No values in it. In this paper, we are going through different steps to finally obtain an accuracy of 88% with good precision and recall values.

Keywords: Attrition rate, Classification algorithms, Decision tree, Random Forest, K Nearest Neighbourhood, Neural Networks, eXtreme Gradient boosting, Ada-Boosting, Regularization parameters, HR dataset and Kaggle.

I. Introduction:

Employee attrition will affect a company in many ways. If any employee leaves an organization, more focus is towards knowledge transfer than any other productive work. Finding a suitable replacement for all kinds of roles may not be easy and sometimes even a lengthy process including the additional training time to improve the skills of the new joiner. The team, from where the employee leaves or about to leave, its performance will get decreased and the work pressure among the team members will also increase.

To avoid such scenarios, the HR team tries different ways to know which employee is not content in the current role. Then, the HR team motivates such employees to avoid the movement. Accurate prediction of employee attrition and taking preventive measures play a valuable role in maintaining the company's productivity. So, an automated approach using machine learning algorithms to predict employee attrition rate has become the essential need for every company.

II. Related Works

From several years onwards, [redacted] reasearches are done on predicting the employee turnover and many new models have been introduced both practically and theoretically. In recent periods, predictions are done using statistical analysis techniques.

Storey [1] stated that success of an organization is dependent on holding employees along with job satisfaction. He observed that lack of job satisfaction or factors regarding the workplace are the main reasons for attrition. Actually, attrition takes place because of several reasons including imbalance in work- personal life, poor knowledge of the field, poor management as well as poor compensation. These factors make the employee to leave from the current job and to look for a better one. At the same time, if problems associated with the working place are solved, the attrition rate decreases notably.

Jantan [2] applied c4.5 Decision tree, Random Forest, Multilayer Perceptron and Radial Basis Functional Network for the employee turnover problem. Based on the results, the authors recommended the C4.5 Decision tree as the most promising approach for the data set they considered.

Nagadevara [3] considered the relationship of withdrawal behaviours like lateness and absenteeism, job content, tenure and demographics on employee turnover and applied Artificial neural networks, logistic regression, classification and regression trees (CART), classification trees (C5.0), and discriminant analysis to predict attrition rate. The authors finally stated that Random Forest with Decision trees offered best performance.

Pankaj Ajit [4] took various algorithms like Logistic Regression, Naïve Bayesian, Random Forest (Depth controlled), SVM (RBF kernel), Linear Discriminant Analysis LDA, KNN (Euclidean distance) and XGBoost to apply on the same data set which we are using in this paper. He found that the XGBoost algorithm performs better among the remaining algorithms he used.

Joao Marcos de Oliveira [5] attempted to predict the employee attrition using Neural Networks, Recurrent Neural Networks and Gated Recurrent Neural Networks. The authors found that Gated Recurrent performs well when the data is balanced and Recurrent Neural Networks performs better when the data is unbiased. As the data we consider for our study is imbalanced in terms of attrition class, we are using Recurrent Neural Networks for the purpose of performance comparison.

Xiang Gao [6] performed employee attrition prediction using Decision tree, Random Forest, Back Propagation Neural Networks and Logistic regression. After comparing the performance of these algorithms based on the ROC curve, the authors stated that Random forest's performance is superior compared to other algorithms.

Yue Zhao [7] published a paper on employee attrition prediction using machine learning algorithms. The machine learning algorithms they used are decision tree, random forest, gradient boosting trees, extreme gradient boosting, a logistic regression, support vector machines, neural networks, linear discriminant analysis, Naïve Bayes method and K-nearest neighbour method. The authors arrived at a conclusion that XGB, Decision Tree, Random Forest and Neural Networks perform well for the dataset they considered.

Rahul Yedida [8] did employee Attrition Prediction in which he compared KNN, Naïve Bias, Logistic Regression and MLP classifier and reported the superiority of KNN classifier in terms of accuracy and predictive effectiveness through the ROC curve observation. When used with its optimal configuration, it is a robust method that delivers accurate results despite the noise in the dataset, which is a major challenge for machine learning algorithms.

Yoon [9] state that the key role of the HR department is to identify the characteristics and needs of every employee personally and some associated information related to employee attrition. The authors made a research among the employees serving in five star hotels and found that job satisfaction is the important factor affecting the turnover. They have done research on how giving priority to an EMC error management culture results in more job satisfaction and less turnover.

Labrague [10] did research on attrition rate prediction among nurses working in Philippines hospitals. The authors concluded that age, job satisfaction, and job stress are the factors mostly affecting attrition. However, they used the Apriori model which is not accurate since extracted features for employee attrition work well in rare cases only.

Amir [11] implemented knowledge discovery steps on real employees data of a manufacturing plant. They chew over many characteristics of employees such as age, technical skills and work experience. They used the Pearson Chi-Square test to find out the importance of data features.

In this paper, we attempted to use six classification algorithms to identify the best fit algorithm for employee attrition problem and to build a model for predicting the future employee attrition rates and predictions. We also used the Ada Boosting model, one of the best ensemble models for unbiased data because in Ada-boosting we take care of both weak trees and trees into consideration by giving the trees with their appropriate weights depending upon their error measures.

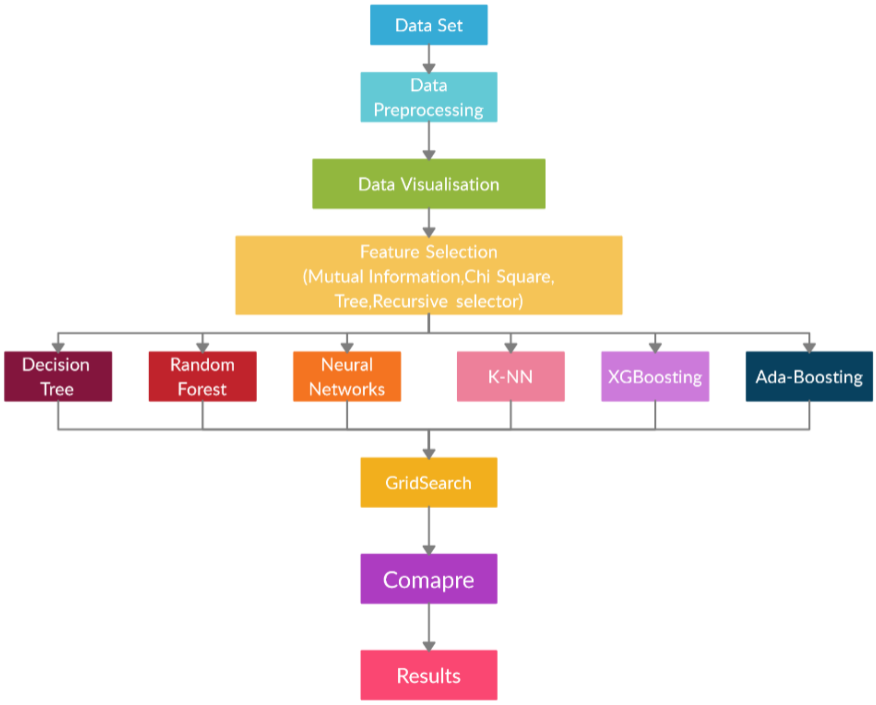
Commented [d1]: Is itmright?

Commented [LG2R1]: Yes mam

III. Methodology

The data set that we consider in this paper is IBM HR Analytics Employee Attrition & Performance data set taken from the Kaggle website [12] .The flow of the methodology is as follows.

New one:



old one:

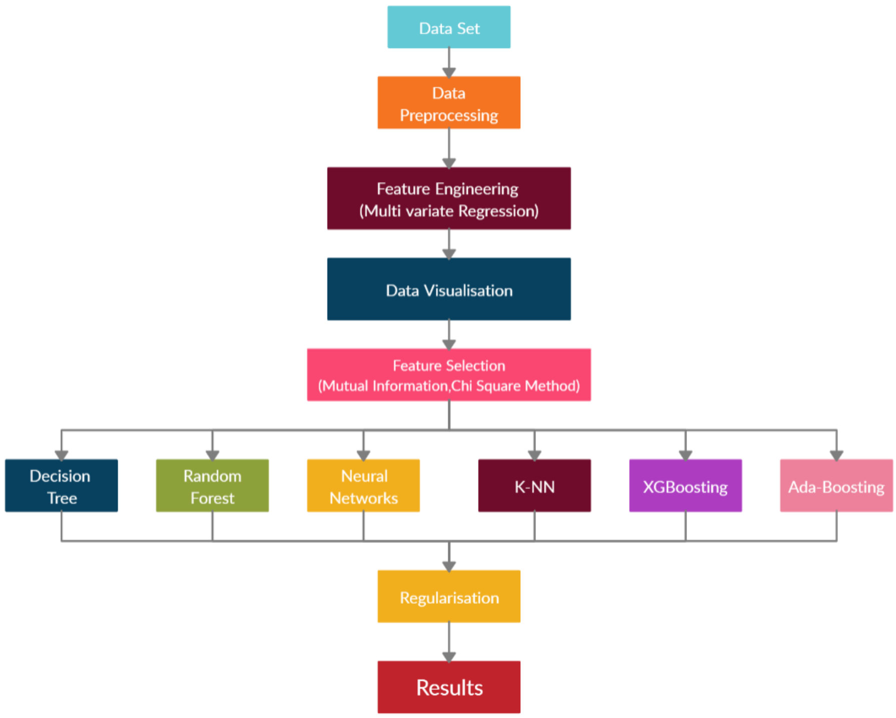


Fig:- Methodology Flow

Data-pre-processing:

The HR-data set from Kaggle contains 35 features in which 34 are independent features and the remaining one is a dependent feature which is our output. Figure.1 lists all the features of the data set along with the possible values each feature can take and then its data type. The features are almost self-explanatory from their names itself.

The features in this data set are of different data types like categorical, binary and numerical. This required us to convert every feature into numerical form to ease the calculations as many models works with only numerical data ,in addition to pre-processing the data. The feature engineering can give a better accuracy score as it removes the unwanted features. To start with, we first converted the categorical data to binary numeric data using an encoding script which uses a dictionary concept.

```
RangeIndex: 1470 entries, 0 to 1469
Data columns (total 35 columns):
Age                         1470 non-null int64
Attrition                   1470 non-null object
BusinessTravel              1470 non-null object
DailyRate                   1470 non-null int64
Department                  1470 non-null object
DistanceFromHome            1470 non-null int64
Education                   1470 non-null int64
EducationField              1470 non-null object
EmployeeCount               1470 non-null int64
EmployeeNumber              1470 non-null int64
EnvironmentSatisfaction     1470 non-null int64
Gender                      1470 non-null object
HourlyRate                  1470 non-null int64
JobInvolvement              1470 non-null int64
JobLevel                    1470 non-null int64
JobRole                     1470 non-null object
JobSatisfaction             1470 non-null int64
MaritalStatus               1470 non-null object
MonthlyIncome               1470 non-null int64
MonthlyRate                 1470 non-null int64
NumCompaniesWorked          1470 non-null int64
Over18                      1470 non-null object
OverTime                    1470 non-null object
PercentSalaryHike           1470 non-null int64
PerformanceRating           1470 non-null int64
RelationshipSatisfaction    1470 non-null int64
StandardHours               1470 non-null int64
StockOptionLevel            1470 non-null int64
TotalWorkingYears           1470 non-null int64
TrainingTimesLastYear       1470 non-null int64
WorkLifeBalance             1470 non-null int64
YearsAtCompany              1470 non-null int64
YearsInCurrentRole          1470 non-null int64
YearsSinceLastPromotion     1470 non-null int64
YearsWithCurrManager        1470 non-null int64
dtypes: int64(26), object(9)
memory usage: 402.1+ KB
```

Figure.1 Features of Kaggle Data Set

~~In this step, we applied multivariate regression to find the importance of each feature to the target feature depending upon its coefficient value and p value which is confidence value. The results are indicated in Figure 2.~~

~~Figure.2 Effect of applying multivariate regression~~

~~We noticed that the features '*Daily Rate*', '*Monthly Rate*' *and* '*Monthly Income*' have values in terms of exponential powers, almost equal to zero and their confidence value which is P value almost nearer to 1 . This concludes that these are contributing very less considerably negligible importance to predict the target feature. So, we remove '*Monthly rate*' and '*Daily rate*' and continue with '*Monthly income*' for further steps.~~

Data Visualization

In this phase, we are plotting different types of graphs including heat maps as shown in Figure.3, correlation graphs , counter plots, histograms and distribution plots among various features as shown in Figure.4 (a-d) to understand the relations between different features to target features.

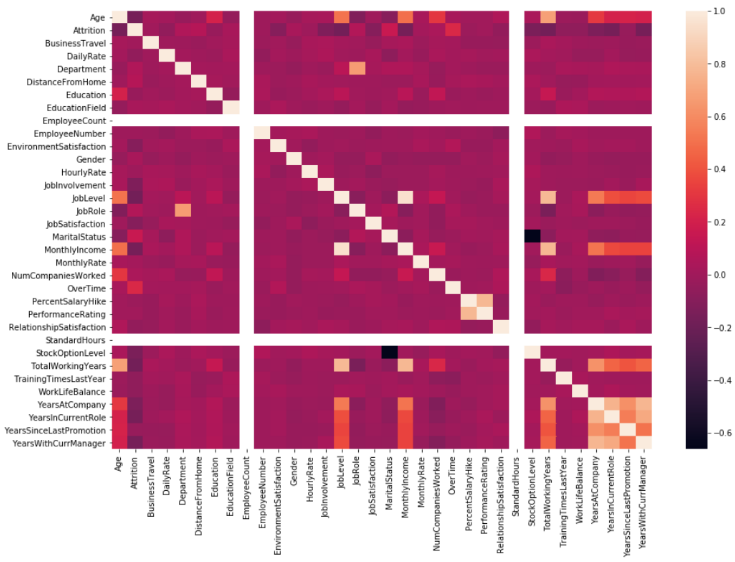


Figure 3. Correlation Heat map of all features

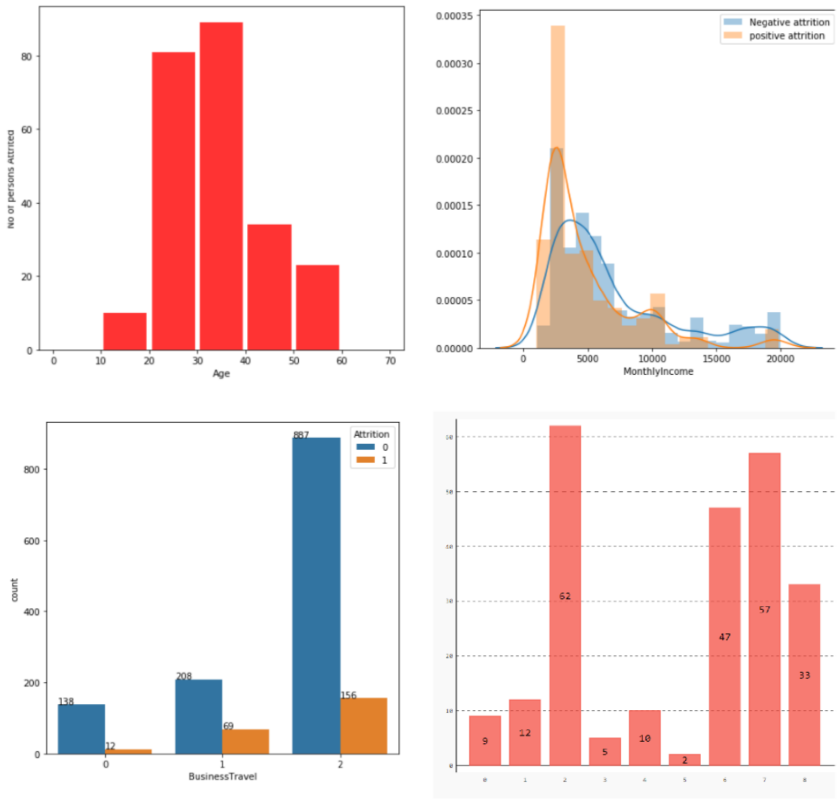


Figure. 4 (a-d) correlation graph , counter plot, histogram and distribution plot

Feature Engineering

In this step , we can divide all the features we are having into two categories , Continuous Numerical and Categorical Data with String values and Ordinal Data.

Commented [d3]: seperate these 4 figures so that we can ne them as a, b, c, d....in right side diagrams x-axis, y-axis name are not there.....Labels are not visible.....If you make as separate diagrams, we can resize them

i) Continuous Numerical Features :- 'Age', 'DailyRate', 'DistanceFromHome', 'Education',' Employee Count', 'Employee id', 'HourlyRate' ,'JobLevel', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours' , TotalWorkingYears', 'TrainingTimesLastYear', 'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion', 'YearsWithCurrManager'.

From the above variables Employee Count, Over 18 and Standard Hours are constant variables they are having constant value for all the entries so these variables doesn't make any sense in the prediction so we removed them. And Employee id is unique for each employee so we removed this one also .

After removing these 4 variables we will scale the remaining variables by using the Standardization scaling formulae

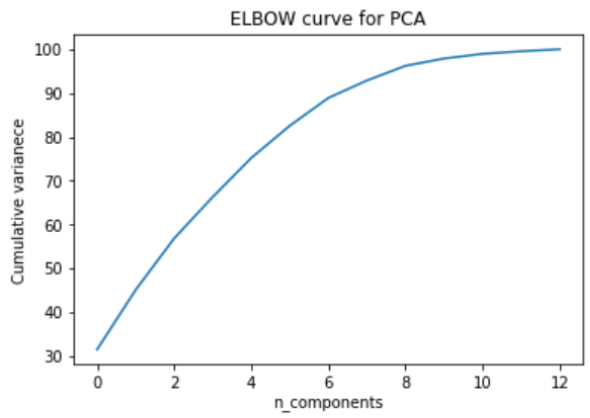
$$Xnew = \frac{Xold - u}{\sigma}$$ μ = mean, σ = standard deviation

ii) Categorical Features (String values): The features 'BusinessTravel', 'Department', 'EducationField', 'Gender', 'JobRole', 'MaritalStatus' and 'OverTime' are categorical features we will do one hot coding on them to convert these variables into Boolean forms. We are doing this because we don't know which variable is Ordinal or Nominal, So we are doing one hot encoding for these and we will remove original features from the data . After doing one hot encoding we will get 51 features in total.

iii) Ordinal Data: we will not do any modifications for these features as they are in their best shape .

PCA(Principal Component Analysis) :

As we know that principal component cannot be performed on categorical variables along with numerical variables we will separate the continuous numerical variables and perform PCA operation on them . There were totally 13 numerical features are there now will draw a elbow curve for them to pick the best number of features without loosing more information.



And the cumulative variance values are `array([ 31.51160882, 45.10839722, 56.86614763, 66.30600138, 75.2075295 , 82.56779817, 88.89630377, 92.87309077, 96.22304125, 97.88298146, 98.95452676, 99.53852271, 100. ]).`

From the above values we observe that the change in variance from 8th to 9th is 3.5 and the information covered is up to 93% we can take 8 features so now we converted the 13 numerical variables to 8 features and keep them a side for some time later we will add them after the feature selection process.

Feature selecting

After applying Mutual Information method, Recursive Method , Tree selector method and Chi square method to all features with respect to attrition, we consider only the top 15 features from each method and we will join them into set . As a result we will get 23 features , 'BusinessTravel_Non-Travel','BusinessTravel_Travel_Frequently','Department_Sales','EducationField_Human Resources','EducationField_Life Sciences','EducationField_Technical Degree','EnvironmentSatisfaction','JobInvolvement','JobRole_Healthcare Representative','JobRole_Human Resources','JobRole_Laboratory Technician','JobRole_Manager','JobRole_Manufacturing Director','JobRole_Research Director','JobRole_Sales Representative', 'JobSatisfaction','MaritalStatus_Divorced','MaritalStatus_Single','OverTime_No', 'OverTime_Yes','RelationshipSatisfaction','StockOptionLevel' and 'WorkLifeBalance'. for this 23 features we will add our 8 Pca features now totally we will get 31 features.

Algorithms Used :

RFE (Recursive feature eliminator or Recursive Selector):

RFE works by searching for a subset of features by starting with all features in the training dataset and successfully removing features until the desired number of features remained.

This is done by fitting the given ml algorithm used in the the model, ranking features by importance, discarding the least important features, and re-fitting the model. This process is repeated until a specified number of features remains.

Features are scored either using the provided machine learning model (eg: Decision tree , Random forest etc ..,)or by using a statistical method.

Tree Selector:

Tree selector works same as RFE but in the feature scoring algorithm it always uses Decision tree as its algorithm. In decision tree it uses either entropy or Gini as its scoring index criteria.

Decision Tree: Decision trees are useful when data possess nonlinear patterns present in it. It is represented as a flow chart in which internal nodes represent the test on a particular-feature and the branches represent output of the test and the child node represents the class in which the input / item is classified. We consider Entropy as the splitting constraint. The decision tree which we get for the employee attrition prediction is shown in Figure 6. The formulae for the Entropy is given by

$$\text{Entropy}(S) = \sum -p_i * \log_2(p_i)$$

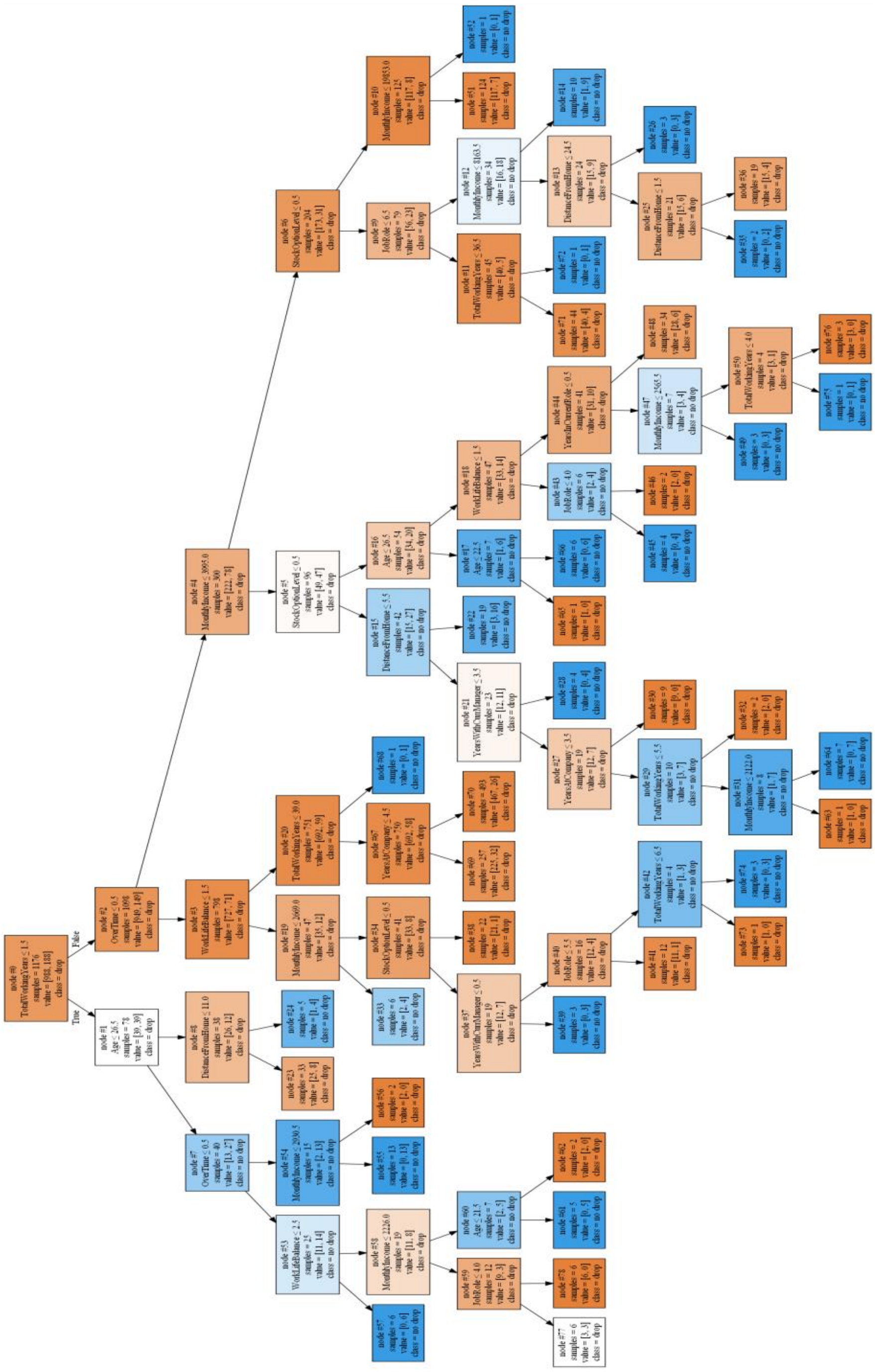


Figure 6 : Decision Tree

Random Forest : The working principle is given in Figure.7 for random forest algorithm It constructs a number of decision trees using CART procedure and the output will be either mean or mode of all trees present in a random forest. This helps the algorithm to avoid overfitting. The diagram representation of the Random forest algorithm is shown below.

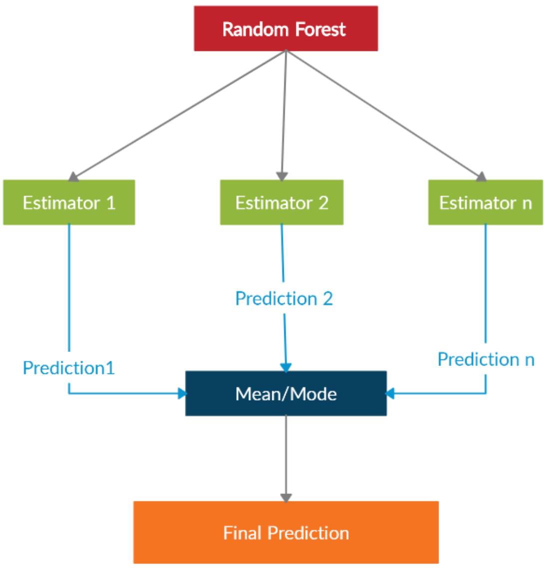


Figure.7 : Illustration of Random Forest Woking Principle

In random forest we use Mean Squared Error to how our data branches from each node.

$$MSE = \frac{1}{N} \sum_i (f_i - y_i)^2$$

yi = Actual data

fi = Predicted data

N = Total number of data points

This formula calculates the distance of each node from the predicted actual value, helping to decide which branch is the better decision for your forest.

While performing random forest we can use either Gini-index or entropy as the criteria for the splitting of node in decision trees. The formulae for the Gini-index is

$$Gini = 1-\sum(p^2)$$

The formulae for the Entropy is

$$Entropy(S) = \sum -p_i * \log_2(p_i)$$

Neural Networks : Neural networks are mostly used for image classifications and computer vision to learn patterns on the unstructured data. As we know that Neural Networks works best for unstructured data, it is capable of taking out difficult patterns from data. Since our data is biased, we selected neural networks to find how best this works to our problem.

The input for each individual neuron can be given by $z=w^Tx+b$ where W^T is weight vector and b is bias. The output of the neuron can be given by the function $y = g(z)$ the function g is called sigmoid function . The loss function can be given by

$$J(W,b) = \frac{1}{m}\sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)})$$
$$L(\hat{y}, y) = -(y \log \hat{y} + (1-y)\log(1-\hat{y}))$$

The sigmoid function is given by

$$G(x) = \frac{1}{1+e^{-x}}$$

K-Nearest Neighbourhood : KNN algorithm works based on geometrical principles. This algorithm plots all the points in the multi-dimensional plain and then indexes them according to their class . When a new point enters, it gets plotted on plain and then takes the mode of the k nearest points as the class of it. It works best for classification models when there exists a geometrical relationship present among the features of the data.

eXtreme Gradient Boost (XGB) : XGB algorithm is an advanced version of Random forest. In random forest, each sub tree is independent but in this model the trees are interlinked and error in the current tree is adjusted through constructing consecutive trees by reducing errors. Every tree has some weight in the final prediction of the model according to its error rate .

The main formulae's we use in the XGB is :

similarity score = $\Sigma\theta$ / ($\ddot{e}$ +λ)

λ *is used for regularization which varies between 0-1*

θ = *error residual= difference between the predicted and expected values*

$\ddot{e}$ = *error samples*

Ada Boost Algorithm: This model is constructed by ensembling the weak learners. Here linear regression is used as the weak leaner for regression models, stumps as the classification models. Stump is a tree with one parent node and two leaf nodes. and work similar to the trees in XGB. Each consecutive stump is created to overcome the errors of previously constructed stump. Adaptive Boosting algorithm performs better on unbalanced data.

Experiment:

Regularization :

For each model, we apply regularisation by changing the parameters and drawing graphs with respect to the accuracy and parameters. After analysing it, we can select the best parameters and include these model parameters for final comparison.

The data set used in this paper contains 35 features in which 32 features are useful for the prediction of attrition not including employee name, S.No and employee_id. From the data visualization step, we identified only 15 features as important. The output variable is a binary variable having value of Yes / No indicating the prediction of employee movement. For our experiment, we used 70% data from the data set for training and 30% for testing. The graphs are as follows:

D-tree Random-Forest

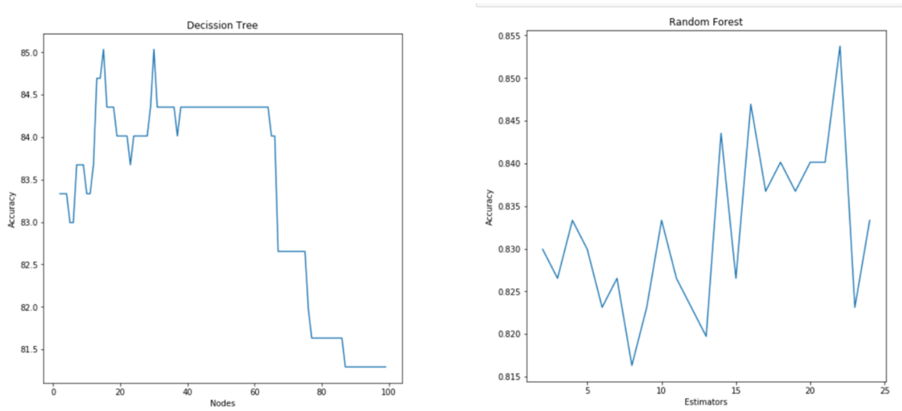


Figure 8. Regularized Parameters Decission Tree vs Random Forest

From Figure 8, we could finalize regularized parameters for D-tree algorithm as 40 nodes and for Random forest regularized value for number of estimators is around 20.

XG Boost :

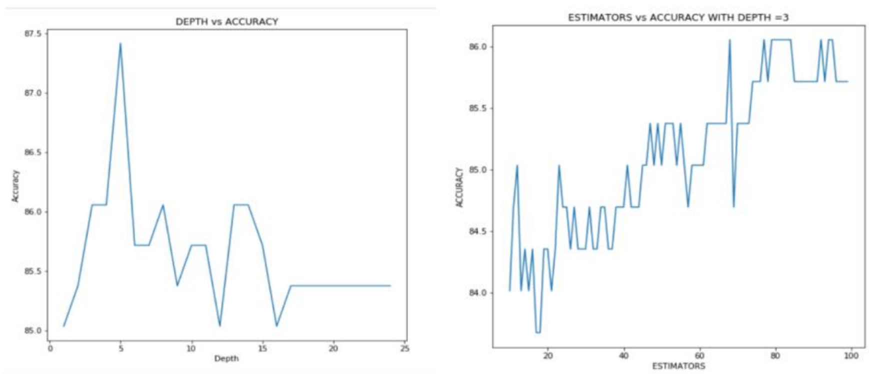
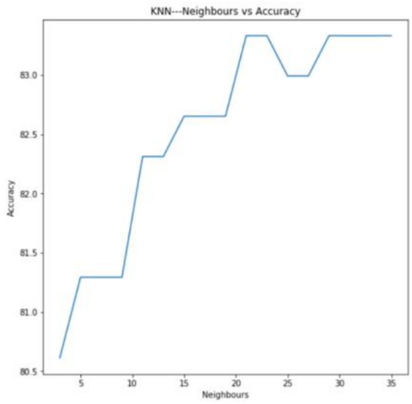


Figure : 9 XGBoost Regularized Parameters

From Figure 9, we take regularized parameters as Depth = 3 and Estimators = 40 for XGBoost algorithm.

Commented [d4]: Give Figure number and give captions

KNN



Ada-Boost

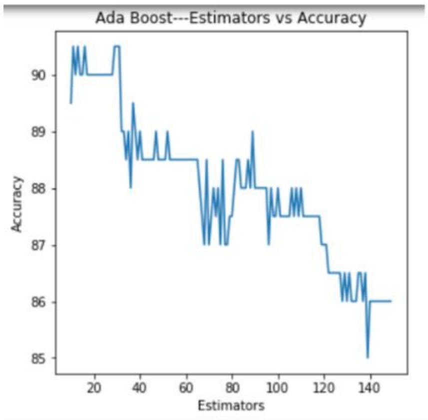


Figure : 10 KNN and Ada-Boost Regularized Parameters

From Figure 10 regarding regularized parameters, for KNN we observed that k's regularized value is 5 since from the graph we observe that the accuracy is same along 5-10. For AdaBoost we can take 20 estimators as the regularised value because we observe that there are very less fluctuations and graphs seems to be constant from 30 - 60, so we took 40 as an regularized value.

From the above graphs we got the regularized parameters for all the algorithms. Now we will use these parameters for the final comparison of all the algorithms.

5) Implementation:

To implement, we used Jupiter IDE of Anaconda platform in 3.8.2. ~~By using multivariate algorithm, Mutual Information and Chi-Square Method, we reduced the features from 35 to 15. We now apply these features to every algorithm and analyse precision score, recall and accuracy.~~

Since our data consists 83% entries of the employees with attrition value as '0' and 17 % entries of the employees with attrition value as '1', we take harmonic mean of each metric like precision, recall and f1 score. When we are using confusion matrix to find the metrics, the resulting values we get are only dependent on the '1' cases. In our model it is about attrition positive employees, but it will not see about the 'zero's'. so now we calculate precision, recall and F1-Score regarding zero's also and we will take hormonic mean of both one's and zero's to get the final precision, average and F1-Score values.

Harmonic mean of a and b = 2*a*b/(a+b)

AUC-ROC for Each Model :-

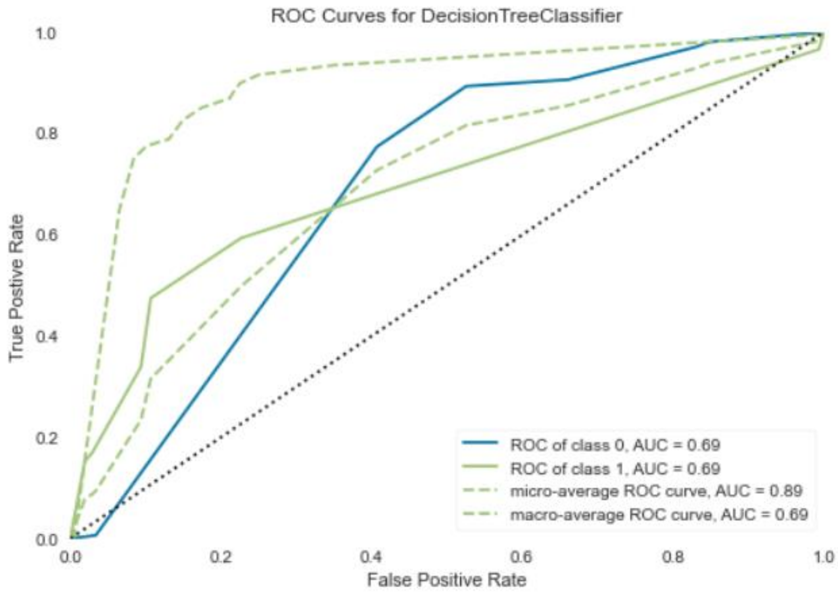
In Machine Learning, performance measurement is an essential task. So when it comes to a classification problem, we can count on an AUC - ROC Curve. When we need to check or visualize the performance of the multi - class classification problem, we use AUC (**Area Under The Curve**) ROC (**Receiver Operating Characteristics**) curve. It is one of the most important evaluation metrics for checking any classification model's performance. It is also written as AUROC (**Area Under the Receiver Operating Characteristics**)

AUCROC curve is an graph between the True Positive rate and False Positive Rate

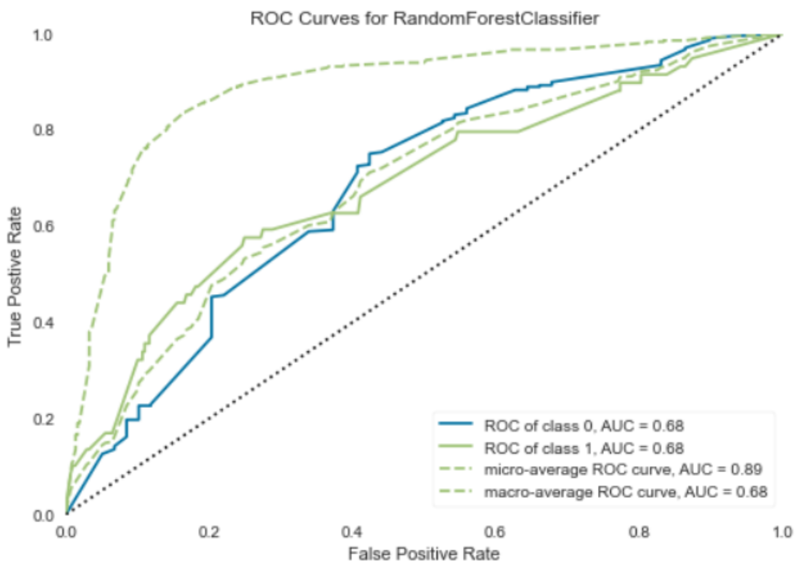
Whereas True Positive Rate $= \frac{True\ positive}{True\ Positive + False\ Negative}$ and

False Positive Rate $= \frac{False\ \ positive}{False\ Positive + True\ Negative}$.

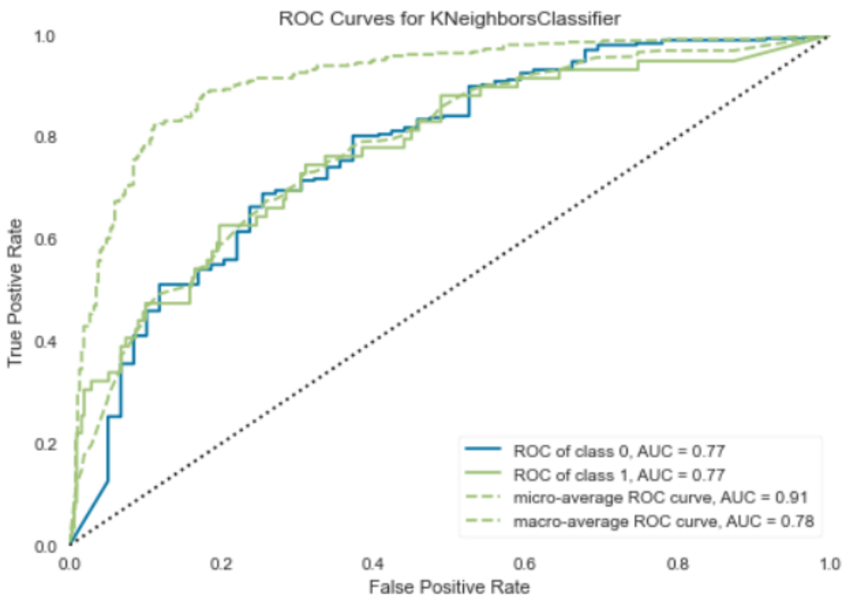
AUCROC of Decission-Tree:-



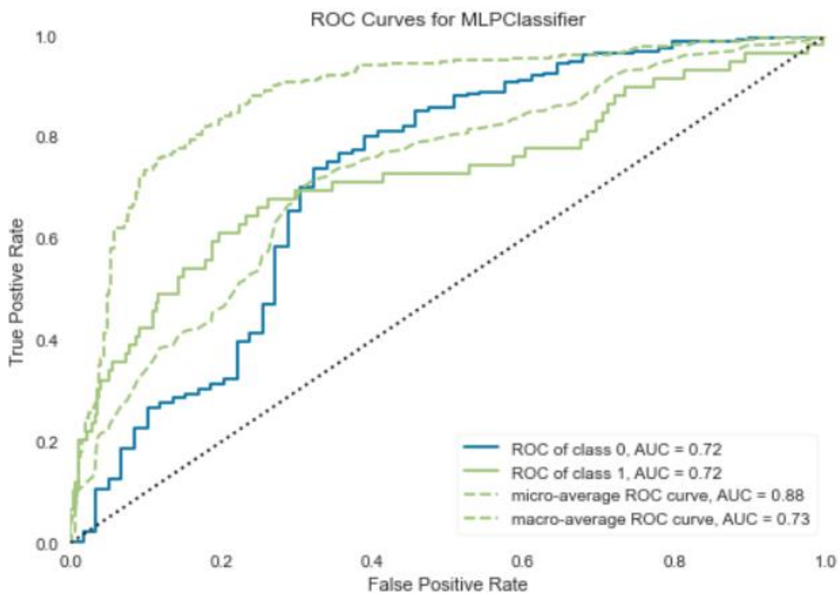
AUCROC of Random Forest:-



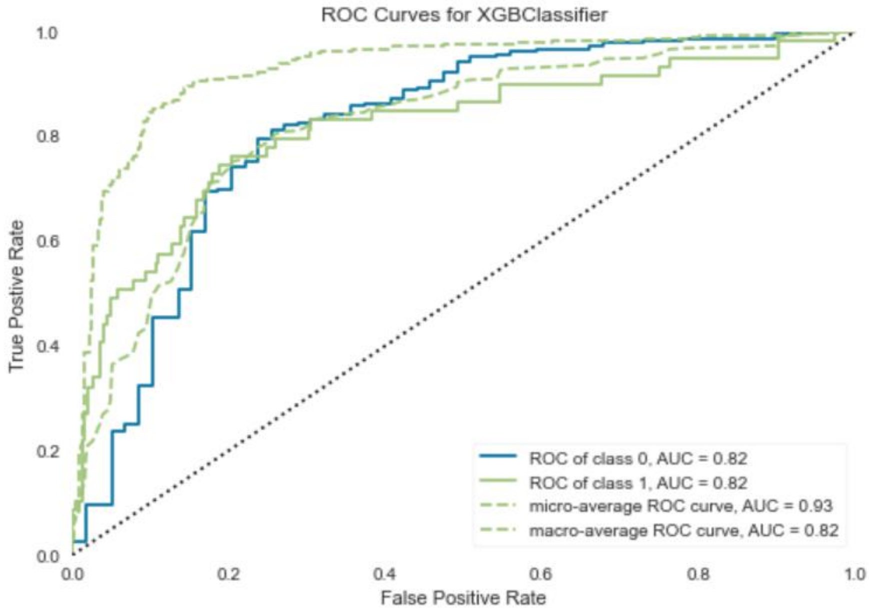
AUCROC of KNN:-



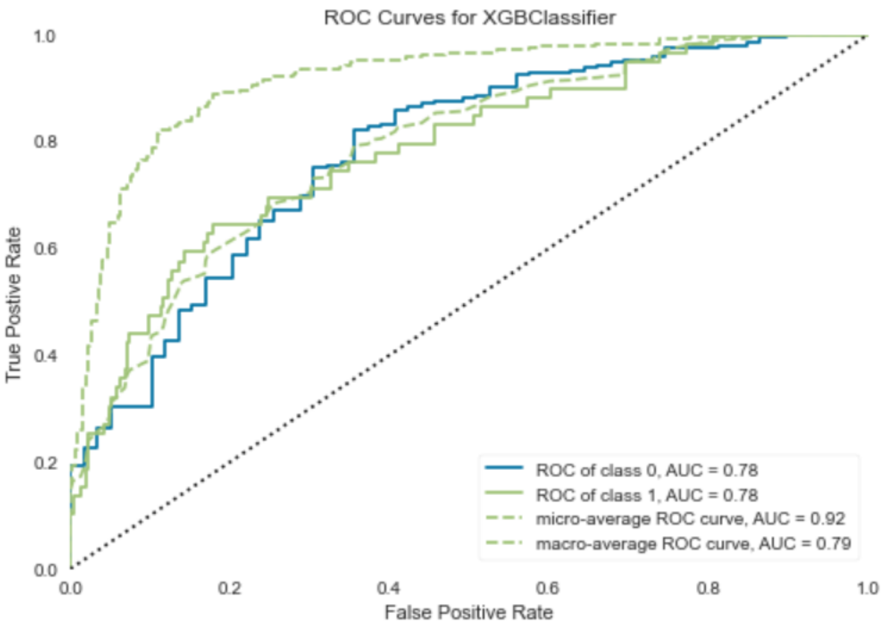
AUCROC of Neural Networks or Multi-Layer Perceptron :-



AUCROC of eXtreme Gradient Boosting :-



AUCROC of Ada Boosting Model :-



6) Results:

The results which we get when we use harmonic mean of classes are as follows:

~~Table 1 : Performance Comparison~~

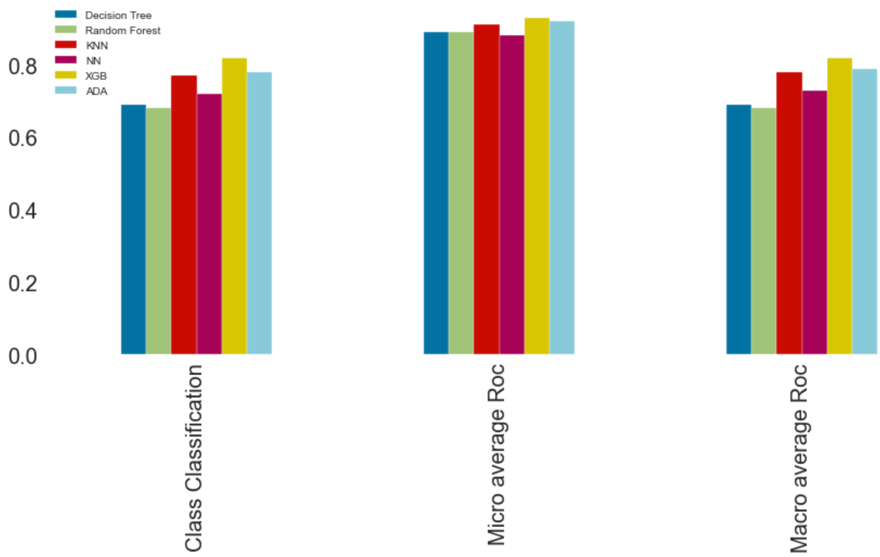
~~Model~~	**~~Avg Precision~~**	**~~Avg Recall~~**	**~~Avg F1 Score~~**	**~~Accuracy~~**
~~D Tree~~	~~0.49~~	~~0.68~~	~~0.56~~	~~84~~
~~Random Forest~~	~~0.33~~	~~0.51~~	~~0.40~~	~~82~~
~~XGB~~	~~0.44~~	~~0.72~~	~~0.54~~	~~85~~
~~K nearest~~	~~0~~	~~0~~	~~0~~	~~83~~
~~Neural Networks~~	~~0.49~~	~~0.69~~	~~0.57~~	~~85~~
~~Ada Boost~~	**~~0.61~~**	**~~0.74~~**	**~~0.67~~**	**~~88~~**

~~From the figure when we see K nearest algorithm, all its metrics zero because KNN algorithm performed well in finding Zero's but failed in classifying into one's correctly. So the overall metric value become zero . If we observe that the KNN model classified every value into zero's because in our data, the 83% target values trained are zero's. So it simplified itself to classify every entry into zero. This is the major problem faced by KNN when data is imbalanced.~~

Commented [LG5]: We can remove it and put zero's value in the table .

From figure we can understood that XGBoost performs best compared to the remaining algorithms with an accuracy of 88% and F1 score of 0.87. Since its Recall score is greater than the Precision score, it says that false negative is greater than false positives. So Type 2 errors are very low. One more observation here is that the precision, recall and f1-score values are little low because we are not giving importance to only classify into one's (attrition positive) but also for zero's (attrition negative).

We compare all the results of the AUCROC in one graph which is as follows :



From the above figure we can observe that in the AUCROC curves also in each criteria the XGBoost performs better than the remaining five algorithms.

7) CONCLUSION :

In this paper, we implemented certain classification algorithms on the dataset taken from Kaggle named IBM HR Analytics & Performance dataset having 35 features for the prediction of employee attrition. We are able to obtain 88% accuracy when we use Ada Boosting algorithm. When we observe our model it is giving best and balancing precision and recall which tells that our model considered both type 1 error and type 2 error reductions.

In future we will try to build an deep learning model with Convolutional Neural network with more data. The dataset we used in this paper consists of around 1500 rows which are less. If we get more data from any research organization or from any competition then our model is regularized very well in the upcoming model. As we know the logic of machine learning is more data, more precision in prediction. Afterward, we can implement deep learning models when we are having more features which are complex but gives more precision in prediction algorithm.

9) REFERENCES :

1) DAVID J. STOREY," Understanding the Small Business Sector: Reflections and Confessions"

2) Jantan, Hamidah Jantan et al. "Human Talent Prediction in HRM using C4.5 Classification Algorithm " (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 08, 2010, 2526-2534.

3) Nagadevara, V., Srinivasan, V. & Valk, R. (2008). Establishing a Link between Employee Turnover and Withdrawal Behaviours: Application of Data Mining Techniques, Research and Practice in Human Resource Management, 16(2), 81-99.

4) Pankaj Ajit, "Prediction of Employee Turnover in Organizations using Machine Learning Algorithms A case for Extreme Gradient Boosting" International Journal of Advanced Research in Artificial Intelligence, Vol. 5, No. 9, 2016.

5) Joao Marcos de Oliveira, Matthäus P. Zylka, Peter A. Gloor, and Tushar Joshi:" Mirror, Mirror on the Wall, Who Is Leaving of Them All: Predictions for Employee Turnover with Gated Recurrent Neural Networks" © Springer Nature Switzerland AG 2019 Y. Song et al. (eds.), Collaborative Innovation Networks, Studies on Entrepreneurship, Structural Change and Industrial Dynamics, https://doi.org/10.1007/978-3-030-17238-1_2 .

6) Xiang Gao , Junhao Wen , and Cheng Zhang:" An Improved Random Forest Algorithm for Predicting Employee Turnover" Volume 2019, Article ID 4140707, https://doi.org/10.1155/2019/4140707

7) Yue Zhao , Maciej K. Hryniewicki , Francesca Cheng, Boyang Fu and Xiaoyu Zhu:" Employee Turnover Prediction with Machine Learning: A Reliable Approach" © Springer Nature Switzerland AG 2019 K. Arai et al. (Eds.): IntelliSys 2018, AISC 869, pp. 737–758, 2019. https://doi.org/10.1007/978-3-030-01057-7_56

8)Rahul Yedida, Rahul Reddy, Rakshit Vahi, Abhilash, Rahul J, Deepti Kulkarni:" Employee Attrition Prediction", https://arxiv.org/ftp/arxiv/papers/1806/1806.10480.pdf

9) H. S. Jung and H. H. Yoon, "Error management culture and turnover intent among food and beverage employees in deluxe hotels: the mediating efect of job satisfaction," Service Business, vol. 11, no. 4, pp. 785–802, 2017.

10) L. J. Labrague, D. Gloe, D. M. McEnroe, K. Konstantinos, and P. Colet, "Factors infuencing turnover intention among registered nurses in Samar Philippines," Applied Nursing Research, vol. 39, pp. 200–206, 2018.

11) Amir Mohammad Esmaieeli Sikaroudi, RouzbehGhousi, Ali EsmaieeliSikaroudi:" A data mining approach to employee turnover prediction (case study: Arak automotive

parts manufacturing)" Journal of Industrial and Systems Engineering Vol. 8, No. 4, pp106-121 Autumn 2015

12) Shikha N. Khera, Divya: "Predictive Modelling of Employee Turnover in Indian IT Industry Using Machine Learning Techniques" Vision 23(1) 12–21, 2019 © 2019 DOI: 10.1177/0972262918821221 journals.sagepub.com/home/vis

13) Ramakrishnan Raman, Sandeep Bhattacharya, Dhanya Pramod, (2018) "Predict employee attrition by using predictive analytics", Benchmarking: An International Journal, https://doi.org/10.1108/ BIJ-03-2018-0083

14) Harlieen Bindra, Krishna Sehgal and Rachna Jain:" Optimisation of C5.0 Using Association Rules and Prediction of Employee Attrition" © Springer Nature Singapore Pte Ltd. 2019 S. Bhattacharyya et al. (eds.), International Conference on Innovative Computing and Communications, Lecture Notes in Networks and Systems 56, https://doi.org/10.1007/978-981-13-2354-6_3

15) Devesh Kumar Srivastava and Priyanka Nair:" Employee Attrition Analysis Using Predictive Techniques" © Springer International Publishing AG 2018 S.C. Satapathy and A. Joshi (eds.), Information and Communication Technology for Intelligent Systems (ICTIS 2017) - Volume 1, Smart Innovation, Systems and Technologies 83, DOI 10.1007/978-3-319-63673-3_35